UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/720,096	11/25/2003	Manoj N. Palat	1509-476	4913

22879 7590 04/30/2008
HEWLETT PACKARD COMPANY
P O BOX 272400, 3404 E. HARMONY ROAD
INTELLECTUAL PROPERTY ADMINISTRATION
FORT COLLINS, CO 80527-2400

EXAMINER	
CHAUHAN, LOREN B	
ART UNIT	**PAPER NUMBER**
2193	

NOTIFICATION DATE	DELIVERY MODE
04/30/2008	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

JERRY.SHORMA@HP.COM
mkraft@hp.com
ipa.mail@hp.com

Office Action Summary	**Application No.** 10/720,096	**Applicant(s)** PALAT ET AL.	
	Examiner LOREN CHAUHAN	**Art Unit** 2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *1/18/2008*.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-11,13-20,22-38,40-45 and 47-50* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-11, 13-20, 22-38, 40-45 and 47-50* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

1. This action is in response to remarks filled on 1/18/2008.

2. Claims 12, 21, 39 and 46 have been cancelled. Claims 1, 4, 11, 13, 15, 20, 25-28, 31, 33-34, 38, 41-43, 45, 47 and 49 have been amended. New claim 50 has been added. Claims 1-11, 13-20, 22-38, 40-45 and 47-50 are pending for examination in this application.

Response to Arguments

3. Applicant's arguments filed on 1/18/2008 have been fully considered but they are not persuasive.

In remarks applicant argues:

> Peterson does not teach locating, calculating and replacing operations or steps occur outside the runtime of an application using the code segment.

Examiner's response:

> Examiner respectfully disagrees. In abstract last five lines Peterson teaches that programmed code that is written for use with position dependent code can be established by supplementing and modifying the position dependent code procedures during compilation and linking (e.g. outside of the runtime of an application); further in col.13, lines 47-63

> Peterson teaches that position dependent code call sequences are executed by loading the called procedure at the address designated by the call sequence, the address at which the called procedure is loaded into memory preferably is determined by the digital data processing system when the calling and called procedures are linked and the position dependent code call sequence is modified to reflect the address at which the called procedure is loaded into memory (e.g. during loading and linking i.e. outside of the runtime of an application).

For the rest of the arguments regarding claims 7, 9, 10, 15-18, 24, 30, 34-36 and 40 are moot in view of applicant's amendment to independent claims 1 and 25, which necessitate new grounds of rejections.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5. Claims 1-3, 5-6, 14, 19-20, 22-23, 25-27, 29, 32-33, 37-38, 44-45 and 47-50 are rejected under 35 U.S.C. 102(b) as being anticipated by Peterson (US Pat. No. 5,504,901).

6. As per claims 1 and 25, Peterson teaches the invention as claimed including a method of enabling execution of data-relative code within a non data-relative environment (Abstract lines 1-5), including the steps of:

i) locating one or more instances of the use of a data-relative offset within a module of a code segment (col. 4, lines 37-42) within the non-data-relative environment (abstract lines 1-3; col. 3, lines 47-50 e.g. position independent code);

ii) calculating a new offset independent of a data segment (col. 7, lines 35-40); and

iii) replacing the data-relative offset with the new offset in the code segment module (col. 7, lines 59-64);

steps (i) to (iii) occur outside of the runtime of the application using the code segment (col. 6, lines 61-65; col.13, lines 47-63; abstract lines 22-27).

7. As per claims 2 and 26, Peterson teaches the invention including the method, wherein the data-relative offset is used for computing the target address of a branch instruction (col. 14, lines 7-10).

8. As per claims 3 and 27, Peterson teaches the invention including the method, wherein the new offset is relative to the instruction pointer (col. 9, lines 1-3).

9. As per claim 5, Peterson teaches the method, wherein steps (i) to (iii) occur during runtime of an application using the code segment (col. 9, lines 30-45).

10. As per claims 6 and 29, Peterson teaches the method, including the step of: iv) executing the modified code segment module (col. 12, lines 62-67).

11. As per claims 13 and 32, Peterson teaches the method, including the step of: v) saving the modified code segment module to non-volatile memory (fig. 3A; col. 5, lines 25-30).

12. As per claims 14 and 33, Peterson teaches the method, wherein the code segment module is within a shared library (col. 18, lines 27-30).

13. As per claims 19 and 37, Peterson teaches the method, wherein the code segment is within an application (col. 6, lines 34-36).

14. As per claims 20 and 38, Peterson teaches the method, wherein the code segment was compiled using a compiler which inserts code that uses data-relative offsets (46, fig. 1; col. 6, lines 39-41).

15. As per claim 22, Peterson teaches the method, wherein a dynamic loader performs all the steps (48, fig. 1, col. 3, lines 55-59; col. 6, lines 60-65).

16. As per claim 23, Peterson teaches the method, wherein the data-relative offset is located in step (i) by back-tracing the target register of the branch instruction (fig. 4, col. 15, lines 17-23).

17. As per claim 44, Peterson teaches a code segment module modified by the method (col. 4, lines 7-9).

18. As per claim 45, Peterson teaches a binary file including a code segment module modified by the method (col. 4, lines 7-9).

19. As per claim 47, Peterson teaches Storage media including computer readable the software for performing the method (14, fig. 1).

20. As per claim 48, Peterson teaches a computer system for affecting the method (fig. 1; col. 5, lines 59-60).

21. As per claim 49, Peterson teaches a memory storing the software (14, fig. 1).

22. As per claim 50, Peterson teaches the system in combination with the non-data relevant environment (col. 1, lines 10-15; e.g. position independent code).

Claim Rejections - 35 USC § 103

23. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

24. Claims 4, 7, 9-11, 28, 30-31, 41 and 43 are rejected under 35 U.S.C. 103(a) as being unpatentable over Peterson (US Pat. No. 5,504,901).

25. As per claims 4 and 28, Peterson teaches the invention substantially including a method, wherein the new offset is calculated using the formula:

new offset = the new offset (col. 10, lines 51-52);

link_time_data_address = the data address during link time (col. 10, lines 53-54);

data_relative_offset = the address relative to the data address (col. 10, lines 55-57);

link_time_module_start = the address of the start of the module of the code segment during link time (col. 11, lines 3-6); and instruction_pointer_offset = the offset, relative to the start of the code segment module, of the instruction that calculates the new address using the instruction pointer (IP) (col. 11, lines 6-10).

26. However, Peterson does not explicitly teach new offset = link time data address + data relative offset -- (link_time_module_start + instruction_pointer_offset).

27. Peterson teaches a method for mathematically calculating memory field that store the address of the procedure being called is locatable by means of a memory offset pointer value that indicates the memory location of the memory field relative to an instruction of the call sequence (col. 10, lines 42-col. 11 lines 1-15).

28. It would be obvious to one of ordinary skill in the art that the mathematical relationship described in Peterson's system is similar to the applicant's mathematical relationship and can be achieved by mathematical manipulation from Peterson's relationship.

29. As per claims 7, 9, 41 and 43, Peterson does not explicitly teach wherein the code segment module is executed on an HP-UX platform and/or non-native platform.

30. However, Peterson teaches wide range of data processing environment can use the invention (col. 13, lines 6-10). Thus it would have been obvious to one of ordinary

skill in the art at the time of the invention was made to recognize the term “any data processing environment” includes HP-UX and/or non-native platform.

31. Therefore it would have been obvious to one of ordinary skill in the art at the time of the invention was made to use the invention of Peterson so that it will extend the applicability of the invention.

32. As per claims 10 and 30, Peterson does not explicitly teach the new offset is an absolute runtime address.

33. However, Peterson teaches calculating call instruction address based on adding instruction offset vale and entry point address register value and executing the instruction on that memory location (col. 11, lines 1-13); thus teaches absolute runtime address for instruction to execute.

34. Therefore it would have been obvious to one of ordinary skill in the art at the time of the invention was made to use the invention of Peterson so that memory overhead can be made lower and achieve higher execution speed of a data processing system.

35. As per claims 11 and 31, Peterson teaches the method, wherein the absolute runtime address is calculated using the formula: wherein:

runtime_addr = the absolute runtime address (col. 10, lines 51-52);

link_time_data_addr = the data address during link time (col. 10, lines 53-54);

data_relative_offset = the address relative to the data address (col. 10, lines 55-57;

link_time_code_segment_start = the address of the start of the code segment during link time (col. 11, lines 3-6); and

run_time_code_segment_start = the absolute address of the start of the code segment during runtime (col. 11, lines 6-10).

36. Peterson does not explicitly teach runtime addr = link time data addr + data relative offset - link_time_code_segment_start + runtime_code_segment._start.

37. Peterson teaches a method for mathematically calculating memory field that store the address of the procedure being called is locatable by means of a memory offset pointer value that indicates the memory location of the memory field relative to an instruction of the call sequence (col. 10, lines 42-col. 11 lines 1-15).

38. It would be obvious to one of ordinary skill in the art that the mathematical relationship described in Peterson's system is similar to the applicant's mathematical relationship and can be achieved by mathematical manipulation from Peterson's relationship. Also, Peterson's system's processor uses the address stored in the procedure entry point register in execution of called procedure (col. 11, lines 33-35).

39. Claims 15-18, 24, 34-36 and 40 are rejected under 35 U.S.C. 103(a) as being unpatentable over Peterson (US Pat. No. 5,504,901) in view of Harman (US Pat. No. 6,970,991).

40. As per claims 15 and 34, Harman teaches the step of: vi) copying the code segment module to modifiable memory (fig. 5E; col. 13, lines 50-55). However, Harman does not explicitly teach wherein step (vi) occurs before step (iii) and wherein step (iii) the data-relative offset is replaced in the copied code segment module.

41. It is a common practice in the art that before replacing any code/data segment, process manipulating that code/data segment is being copied to another memory region so that process can recover particular code/data if need to restore the previous state of the application.

42. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention was made to combine the teachings of Peterson and Hartmann so that a memory region copied the code to another memory region before replacing with another content on that memory region.

43. As per claims 16 and 35, Harman teaches the step of vii) allocating the modifiable memory read, write and execute permissions (531, fig. 5E; col. 10, lines 5-9; col. 7, lines 54-64). However, Harman does not explicitly teach wherein step (vii) occurs

before step (vi). Although it is known in the art that in order to copy a code/data segment from memory, accessing process should have read/write permission. Therefore it would have been obvious to one of ordinary skill in the art at the time of the invention was made to use the invention of Harman so that process accessing memory region would have right permission.

44. As per claim 17, Harman teaches the method, wherein the code segment within the shared library is mapped as writable (col. 9, lines 27-31; col. 7, lines 54-64).

45. Therefore it would have been obvious to one of ordinary skill in the art at the time of the invention was made to use the invention of Harman so that process accessing memory region would have right permission.

46. As per claims 18 and 36, Harman teaches the method, wherein the code segment module is one selected from the set of an .init section from the shared library and a .fini section from the shared library (col. 9, lines 34-39; col. 17, lines 49-51).

47. As per claim 24 and 40, Harman teaches using UNIX environment with shared library (col. 7, lines 24-26; col. 6, lines 46-49). Thus it would have been obvious to that LINUX code segment can be used with shared library since LINUX is derived from the UNIX.

48. Therefore it would have been obvious to one of ordinary skill in the art at the time of the invention was made to combine the teachings of Peterson and Harman so that applicability of the invention can widened by making it compatible with LINUX environment.

49. Claims 8 and 42 are rejected under 35 U.S.C. 103(a) as being unpatentable over Peterson (US Pat. No. 5,504,901) in view of Lee (US Pat. No. 5,828, 884).

50. As per claims 8 and 42, Peterson does not explicitly teach wherein the code segment is compiled for a little-endian system and the code segment module is executed on a big-endian system.

51. Lee teaches the wherein the code segment is compiled for a little-endian system and the code segment module is executed on a big-endian system (see Byte Swapping Device on col. 7-col. 8).

52. It would have been obvious to one of ordinary skill in the art at the time of the invention was made to combine the teachings of Peterson and Lee because it will improve the system which allows software developers to develop more efficient, portable and bug-free code with respect to byte ordering issues (Lee col. 3, lines 48-50).

Conclusion

53. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to LOREN CHAUHAN whose telephone number is 571-270-1554. The examiner can normally be reached on Mon.-Thr. 9:30-5:00 (EST).

If attempts to reach the examiner by telephone are unsuccessful, the examiner’s supervisor, Lewis Bullock can be reached on 571-272-3759. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Lewis A. Bullock, Jr./
Supervisory Patent Examiner, Art Unit 2193

Loren Chauhan
Examiner
Art Unit 2193

/Loren Chauhan/
Examiner, Art Unit 2193